ANEXO A

ANEXO A
MENSAJES SIP

Un mensaje SIP es una solicitud de un cliente a un servidor, o una respuesta de un servidor a un cliente.

SIP es un protocolo basado en texto y utiliza el juego de caracteres UTF-8 [1].
Para realizar la interconexión con otras redes es necesario realizar una serie de peticiones para la adquisición de los servicios requeridos, y así obtener el permiso para el tránsito de los paquetes de datos hasta el origen de la solicitud. A continuación se mostrará cómo se realizan las peticiones y la explicación de sus componentes:

A. 1. Peticiones

Las peticiones SIP son distinguidas por tener una línea de petición para una línea de salida. Una línea de petición contiene el nombre del método, una solicitud URI, y la versión del protocolo separados por un simple espacio de caracteres.

La línea de petición termina con CRLF (línea de alimentación de retorno de transporte). Ni CR o LF están permitidos a excepción de la secuencia CRLF al final de la línea. No se permiten espacios en blanco lineales (LWS) en cualquiera de los elementos [2].

Solicitud de línea = Método SP Petición URI Versión SIP CRLF

- Métodos:

  Esta especificación define 6 métodos, REGISTER para el registro de la información del contacto, INVITE, ACK y CANCEL para el establecimiento de periodos de sesiones, BYE para poner fin a las sesiones y OPTIONS para la consulta de servidores en sus capacidades.

- Petición URI:

  La petición URI es un SIP o un URI general. Este indica el usuario o servicio al que esta solicitud está abordando. La petición URI no debe contener espacios o caracteres de control sin escape y no debe estar entre “<>”.
  Los elementos SIP soportan peticiones URI con un régimen distinto de “sip” o “sips”, por ejemplo el esquema URI “tel” de RFC 2806. Los elementos SIP pueden traducir URIs sin SIP utilizando cualquier mecanismo a su disposición, resultando en URI SIP, URI SIPS o algún otro esquema [3].

La petición URI será un URI SIP, como se define en [RFC 3261] o un URI “tel”, tal como se definen en el [RFC 3966]. La solicitud URI en un INVITE inicial para una llamada básica de teléfono se identificará la persona llamada usando un URI “tel” o utilizando la sintaxis telefónica de abonado (es decir, el número de teléfono marcado) en un URI SIP. Cuando la Solicitud URI es un URI SIP, la parte del equipo de la petición URI identificará el SCF o la entidad a la que se dirige el mensaje. La solicitud URI para las solicitudes de otro tipo relacionadas con una llamada telefónica básica se identificar el host objetivo utilizando la dirección IP o FQDN, tal como lo indicó el encabezado de contacto.

- Versión SIP:

  Ambos mensajes de solicitud y respuesta incluidos en la versión de uso de SIP y siguiendo respecto a la versión de pedidos, los requisitos de cumplimiento y la actualización de los números de versión. Para cumplir con esta especificación, las solicitudes de envío de mensajes de SIP deben incluir una versión SIP de "SIP/2.0". La cadena de versiones SIP distingue entre mayúsculas y minúsculas, pero las implementaciones deben enviarse en mayúsculas.
  A diferencia de HTTP/1.1, SIP trata el número de versión como una cadena literal. En la práctica, esto no debería hacer ninguna diferencia.

Method	Send	Recv	Reference
ACK	M	M	See clause 10.2.1.7.1
BYE	M	M	See clause 10.2.1.7.1
CANCEL	M	M	See clause 10.2.1.7.1
INVITE	M	M	See clause 10.2.1.7.1
OPTIONS	O	O	See clause 10.2.1.7.1
REGISTER	O	O	See clause 10.2.1.7.1

Tabla 1. Métodos RFC 3261

Method	Send	Recv	Reference	RFC
UPDATE	M	M	See clause 10.2.1.7.1	RFC 3311
PRACK	M	M	See clause 10.2.1.7.1	RFC 3262

Tabla 2. Métodos extendidos

También es importante para la interconexión de redes establecer unos campos que sirven como registro con la red visitada, mostrando a continuación los campos de cabecera fundamentales para poder iniciar la conexión y transferencia de datos:

A. 2. Campos de cabecera

Los campos de cabecera SIP son similares a los de HTTP, tanto en la sintaxis como en la semántica. En particular, los campos de cabecera SIP siguen las definiciones [H4.2] de sintaxis de la cabecera del mensaje y las normas para ampliar los campos de cabecera en varias líneas. Sin embargo, este último se especifica en HTTP con espacios en blanco implícitos y plegados. Esta especificación se ajusta al RFC 2234 y sólo utiliza los espacios en blanco explícito y plegado como parte integrante de la gramática [4].

Los siguientes campos de cabecera son obligatorios al realizar la petición de interconexión:

- Accept:

  El campo de encabezado Accept sigue la sintaxis definida en [H14.1]. La semántica es idéntica, con la salvedad de que si no hay campo de la cabecera presente Accept, el servidor debe suponer un valor predeterminado de aplicación/sdp.

- Allow:

  El campo de cabecera Allow lista el conjunto de métodos soportados por el UA generando el mensaje.
  Todos los métodos, incluidos los ACK y CANCEL, entendida por el UA deben ser incluidos en la lista de métodos en el campo de cabecera Allow, cuando está presente. La ausencia del campo Allow no debe interpretarse en el sentido de que el UA de enviar el mensaje no admite ningún método, más bien, implica que la UA no está proporcionando ninguna información sobre cuáles son los métodos que soporta.

  El suministro de un Allow en las respuestas a los métodos distintos de las opciones reduce el número de mensajes que se necesita. Ejemplo: INVITE, ACK, OPTIONS, CANCEL, BYE.

- Authentication:

  El campo de encabezado Authentication proporciona Información para la autenticación mutua con revisión HTTP. Un UAS debe incluir este campo de encabezado en una respuesta 2xx a una solicitud que fue autenticado correctamente utilizando la revisión basado en el campo de encabezado de Authentication.

  Los tipos de respuesta para este campo son los siguientes [5]:
  - 1XX. Información (Ejemplo 181)
  - 2XX. Éxito

- 3XX. Redirección
- 4XX. Fallo en la petición, error de cliente.
- 5XX. Fallo de servidor
- 6XX. Fallo global.

- Autorization:

  El campo de encabezado Autorization contiene las credenciales de autenticación de un UA.
  Este campo de encabezado, junto con Proxy-Authorization, rompe las reglas generales sobre varios valores cabecera. Aunque no es una lista separada por comas, este nombre de campo de encabezado puede estar presente en múltiples ocasiones, y no deben ser combinados en una sola línea de cabecera con las normas usuales [6].

- Call-ID:

  El campo de cabecera Call-ID identifica de forma única una invitación particular o todos los registros de un cliente en particular. Una conferencia multimedia sola puede dar lugar a varias llamadas con diferentes Call-ID, por ejemplo, si un usuario invita a una sola persona varias veces a la misma (de larga duración) conferencia. Call-ID distinguen entre mayúsculas y minúsculas y se limitó a comparar byte a byte.

- Contact:

  Un campo de cabecera Contact ofrece una URI, cuyo significado depende del tipo de petición o la respuesta que está en él. Puede contener un nombre para mostrar, una URI con parámetros URI y parámetros de cabecera.
  Estos parámetros se utilizan solamente cuando el contacto está presente en una solicitud de REGISTER o de respuesta, o en una respuesta 3xx.

- Content – Length:

  El campo de cabecera Content-Length indica el tamaño del cuerpo de mensaje, en número decimal de octetos, enviado al destinatario. Las aplicaciones deben utilizar este campo para indicar el tamaño del cuerpo de mensaje para ser trasladado, sin importar el tipo de medio de la entidad. Si un protocolo basado en flujo (como TCP) se utiliza como transporte, el campo de encabezado debe ser utilizado.
  El tamaño del cuerpo del mensaje no incluye la separación de CRLF de campos de cabecera y el cuerpo. Cualquier Content-Length mayor o igual a cero es un valor válido. Si ningún cuerpo está presente en un

mensaje, el valor de campo de cabecera Content-Length debe ser puesto a cero.

- Content – Type:

  El campo de cabecera Content-Type indica el tipo de medio del cuerpo de mensaje enviado al destinatario. Los elementos de "tipos de medio" se definen en [H3.7]. El campo de cabecera Content-Type debe estar presente si el cuerpo no está vacío. Si el cuerpo está vacío, y un campo de cabecera Content-Type está presente, indica que el cuerpo del tipo específico tiene longitud cero (por ejemplo, un archivo de audio vacío).

- CSeq:

  Un campo de cabecera CSeq en una solicitud contiene un solo número de secuencia decimal y el método de solicitud. El número de secuencia debe ser expresable como un entero de 32 bits sin signo. La parte método de CSeq es sensible entre mayúsculas y minúsculas. El campo de cabecera CSeq sirve para ordenar las transacciones dentro de un cuadro de diálogo, para proporcionar un medio para identificar de forma única las operaciones, y para diferenciar entre las nuevas solicitudes y las retransmisiones de petición. Dos campos de cabecera CSeq se consideran iguales si el número de secuencia y el método de solicitud son idénticos.

- From:

  El campo de cabecera From indica el iniciador de la petición. Esto puede ser diferente del iniciador del diálogo. Las solicitudes enviadas por el destinatario a la persona que llama utiliza la dirección de destinatario en el campo de cabecera From.
  El facultativo "display-name" está destinado a ser prestado por una interfaz de usuario humana. Un sistema debería utilizar el nombre para mostrar "Anónimo" si la identidad del cliente debe permanecer oculta. Incluso si el "display-name" está vacío, la forma "name-addr" debe ser usada si el "addr-spec" contiene una coma, signo de interrogación o punto y coma.

- Proxy – Require:

  El campo de encabezado Proxy-Require se emplea para indicar características proxy sensibles que deben ser apoyadas por el proxy.

- Require:

  El campo de cabecera Require es utilizado por UACs para decir a las UASs acerca de las opciones que la UAC espera que la UAS para apoyar con el fin de procesar la solicitud. A pesar de un campo de cabecera opcional, la solicitud no se debe ignorar si está presente. El campo de cabecera Require contiene una lista de etiquetas de opciones. Cada etiqueta de opción define una extensión de SIP que debe entenderse en el proceso de la solicitud. Con frecuencia, esto se utiliza para indicar que un conjunto específico de campos de cabecera de extensión deben ser entendidos. Un UAC conforme con esta especificación sólo debe incluir etiquetas de opción correspondiente a las normas RFC.

- Route:

  El campo de cabecera Route se utiliza para forzar el enrutamiento de una solicitud a través del conjunto de proxies enlistados.

- Supported:

  El campo de cabecera Supported enumera todas las extensiones con el apoyo de la UAC o UAS.
  El campo de cabecera Supported contiene una lista de etiquetas de opción, que son comprendidos por el UAC o UAS. Una UA conforme con esta especificación sólo debe incluir etiquetas de opción correspondiente a las normas RFC. Si está vacío, significa que no se admiten prórrogas.

- To:

  El campo de encabezado especifica el destinatario lógico de la solicitud. El facultativo "display-name" está destinado a ser prestado por una interfaz hombre-usuario. La etiqueta de "tag" sirve como un mecanismo general para la identificación de diálogo.

- Unsupported:

  El campo de cabecera Unsupported enumeran las características que no son soportados por la UAS.

- Via:

  El campo de cabecera Via indica el camino elegido por la solicitud e indica el camino que debe seguirse en las respuestas de enrutamiento. El parámetro de identificación de rama en el campo de cabecera Via

sirve como un identificador de transacción, y es utilizado por los proxies para detectar bucles.
Un campo de cabecera Via contiene el protocolo de transporte utilizado para enviar el mensaje, el nombre de host del cliente o la dirección de red, y posiblemente el número de puerto en el que desea recibir las respuestas. Un campo de cabecera Via también puede contener parámetros como "maddr", "ttl", "recieved", y "branch".
Para implementaciones conformes con esta especificación, el valor de los parámetros de la rama debe comenzar con la cookie mágica "z9hG4bK".

En la tabla 3 se muestran todos los cuadros de cabecera que se establecen para la interconexión entre redes, asignando prioridades en algunos de ellos:

Header	Send	Recv
Accept	O	O
Accept-Encoding	O	O
Accept-Language	O	O
Alert-Info	O	O
Allow	M	M
Authentication-Info	O	O
Authorization	O	O
Call-ID	M	M
Call-Info	O	O
Contact	M	M
Content-Disposition	O	O
Content-Encoding	O	O
Content-Language	O	O
Content-Length	M	M
Content-Type	M	M
CSeq	M	M
Date	O	O
Error-Info	O	O
Expires	O	O
From	M	M
In-Reply-To	O	O
Max-Forwards	O	O
Min-Expires	O	O
MIME-Version	O	O
Organization	O	O
Priority	O	O
Proxy-Authenticate	O	O
Proxy-Authorization	O	O
Proxy-Require	M	M
Record-Route	O	O

Reply-To	O	O
Require	M	M
Retry-After	O	O
Route	M	M
Server	O	O
Subject	O	O
Supported	M	M
Timestamp	O	O
To	M	M
Unsupported	M	M
User-Agent	O	O
Via	M	M
Warning	O	O
WWW-Authenticate	O	O

Tabla 3. Campos de cabecera

BIBLIOGRAFIA

[1] Internet Engineering Task Force, "*SIP: Session Initiation Protocol*" *IETF* RFC 3261, 2002. [Online] Available: http://www.ietf.org [Accessed: April. 26, 2010].

[2] Internet Engineering Task Force, "The Early Session Disposition Type for the Session Initiation Protocol SIP", IETF RFC 3959, 2004. [Online] Available: http://www.ietf.org [Accessed: Mar. 30, 2010].

[3] International Telecommunication Union / Alliance for Telecommunications Industry Solutions, "Next Generation Technology and Standardization", ITU / ATIS Workshop. Las Vegas, USA, Marzo Del 2006

[4] International Telecommunication Union, "Signalling requirements and protocols for the NGN Service and session control protocols: NGN NNI signalling profile (protocol set 1)", ITU-T Recommendation Q.3401, 2007. [Online] Available: http:// www.itu.int. [Accessed: June. 24, 2010]

[5] International Telecommunication Union, "Signalling requirements and protocols for the NGN Service and session control protocols: NGN NNI signalling profile (protocol set 1); Amendment 1: Extensions of NGN NNI signalling profile including video and data services", ITU-T Recommendation Q.3401 Amendment 1, 2008. [Online] Available: http:// www.itu.int. [Accessed: June. 24, 2010]